Server 12 of system 10 can also be in communication, via internet service providers, such as providers 30a, 30b and 30c and the internet with a displaced or remote computer system 34 coupled to server 34a and display 34b.

Alternately, or concurrently, a user at computer 18, via server 12 can be in communication with a remote server 38 coupled to one or more output devices 38a. These could include printers, film output devices or the like for purposes of producing hard copy, multi-color output documents by printers or printing presses.

It will be understood that the architecture of the system 10 of Fig. 1 is exemplary and variations thereon come within the spirit and scope of the present invention.

Fig. 2 illustrates in more detail various characteristics of the system 10. Common identification numerals, relative to Fig. 1, have been used in connection with corresponding elements of Fig. 2. Fig. 3 illustrates further details of a document importing process in accordance with the present invention and further supplements the disclosure of Fig. 2 .

With respect to Fig. 2, 3 a local user at computer 18 can initiate a document import process, step 100. In a step 102, documents to be imported are translated to a standard, predetermined format, or language, for example such as PostScript, to provide a common input for subsequent processing.

The translated documents are parsed and tagged, step 104, to a plurality of objects and relationships relative to one another. The parsed objects and relationships can then be analyzed and compared taking into account the objects and relationships derived from other documents which have been parsed in a concurrent multiple document batch-mode input process.

The objects and relationships can also be analyzed and compared by the objects and objects relationship manager software 50 in accordance with objects and relationships derived from previously imported documents contained in the system's archive 14. Objects and relationships in the import repository can be analyzed and compared in accordance with user-established rules and standards for purposes of clarification and differentiation of object and object relationships. The objects and relationships in the input repository 52 can also be analyzed and

compared using user-established rules and standards pertaining to object and relationship integrity and accuracy. Additionally, objects and relationships in the input repository 52 can be analyzed and compared to objects previously entered into the archive 14 in accordance with user-established rules to minimize redundant objects and object relationships which are stored in the archive 14.

By eliminating redundancy in the archive 14, system operating efficiency will be improved, storage costs will be reduced and a one-to-many editing process can be implemented wherein a singular linked object, common to many documents or files, can be edited once and have the consequence of the editing process propagate through all of the linked documents and files. The one-to-many editing capability substantially reduces effort needed to up-date files which represent packages or packaging manuals or the like as would be understood by those of skill in the art.

Prior to entry into the archive 14, the User Objects and Relationships Management Interface 54 can provide hard copy reports or visual displays of objects and elements for users to carry out real time editing thereof. Users as a result can manually reconcile objects being imported with previously imported objects in the archive or with user-established rules or standards. Alternately, the process can be carried out automatically. Finally, reconciled objects and object relationships can be moved from the importing objects and relationships repository 52 to the document objects archive 14.

With respect to Figs. 3, 4, the system 10, 10' stores objects and object relationships utilizing a multi-element data structure which incorporates element specific metadata 60a, element properties 60b and element property values 60c. Additionally, the data structure incorporates document-specific data and metadata 62a, document properties 62b and document property values 62c. Tags associated with object oriented data structures by the parser and tagger 104 are carried therewith and enable linking and retrieval of object oriented data structures from archive 14.

Linked objects and relationships can be assembled and compiled using export compiler 106 to create compiled exported documents or files in standard

formats or languages such as PostScript-type languages if desired. It will be understood that the present object-oriented data structures and relationships, stored in archive 14, provide for efficient content management in accordance with the system's one-to-many editing capability.

With respect to the various data structure, elements 60a, b, c and 62a, b, c when stored in the archive 14, relationships between the various components are maintained. System operators are thus able to view and manipulate the document components either within the context of a composite document or a separate document elements. As noted above, industry standard file formats are converted to an efficient object oriented data structure for storage in archive 14.